SYNTHETIC HYPERSPECTRAL DATA CUBES FOR COMPLEX THERMAL SCENES

Christoph C. Borel, William B. Clodius and Pierre V. Villeneuve

Nonproliferation and International Security Division
Space and Remote Sensing Sciences Group, NIS-2, MS C323,
Los Alamos National Laboratory
Los Alamos, NM 87545, USA

ABSTRACT

Generating synthetic hyperspectral data cubes for the thermal is useful because it is very expensive to design and fly such sensors. Using synthetic data is useful in performing trade-off studies, e.g. spectral and radiometric performance requirements, and also in testing new algorithms, e.g. temperature emissivity separation. We have developed a method to simulate complex thermal scenes under variable solar illumination, different materials and including gas plumes. The IDL-based scene simulation tool consists of public domain tools for: 3-D geometry generation, raytracing, spectral and thermal property libraries, atmospheric transmission and emission modeling. The data can then be used in standard hyperspectral processing programs or processed by special purpose programs. We show how this model can be used to generate synthetic data cubes for dispersive and Fourier transform infrared spectrometers (FTIR). The data in turn is then used to evaluate algorithms to separate temperature and emissivity.

Keywords: Hyperspectral, scene simulation, gas plume detection, temperature and emissivity separation

1. INTRODUCTION

We are interested in algorithms to retrieve temperature and emissivity in the long-wave thermal region (8-13 μm). We have developed an iterative temperature emissivity separation (TES) algorithm for hyperspectral sensors (Borel, 1998). To test our algorithm we used synthetic data (as described in the third section) for the following reasons: (1) we can compare the retrieved emissivity to the truth, (2) we can assume that the sensor's spectral and radiometric performance is optimal and (3) we can perform sensitivity studies by assuming errors in the sensor's performance and modeling of the atmosphere which are useful in (a) determining the retrieval errors for actual sensors and (b) to define sensor specifications (e.g. signal to noise ratios and spectral resolution) to meet a certain performance goal (e.g. location plumes, identifying gases,...).

2. SYNTHETIC HYPERCUBE GENERATION

2.1. Geometry Model

A 3-D CAD program called AC3D * was used to generate a complex scene with several concrete buildings with smoke stacks, metal roofs, trucks, loading dock, trees, grass. The AC3D interface is shown in Fig. 1. The AC3D model allows the export of geometry files for rendering programs such as POV, RenderMan, VRML, etc.

The facility was then raytraced using a text-based public domain raytracer *Persistence of vision* (POV) † for a given solar illumination direction. Several different raytracing runs were made:

1. Raytracing the scene using no illumination source and distinct color settings for each object representing different materials results in an image which can be converted to a materials map $M(x, y)$ by simple classification of the red, green and blue channel images produced by the raytracer. A sample is shown in Fig. 2 on the left.

Further author information: Telephone: (505)-667-8972 Fax: (505)-667-3815,
Email : cborel@lanl.gov WWW: http://nis-www.lanl.gov/ borel

*AC3D was written by Andy Colebourne (andy@comp.lancs.ac.uk) and available as share-ware on the WEB at *www.comp.lancs.ac.uk/computing/users/andy/ac3d.html*

†POV 3.0 is available on the WEB at *www.povray.org*

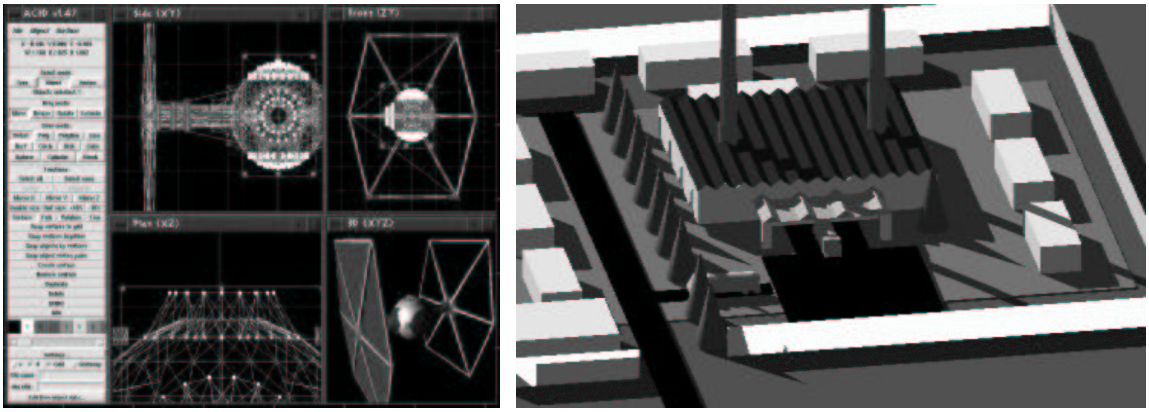

Figure 1. AC3D modeling interface and a sample scene

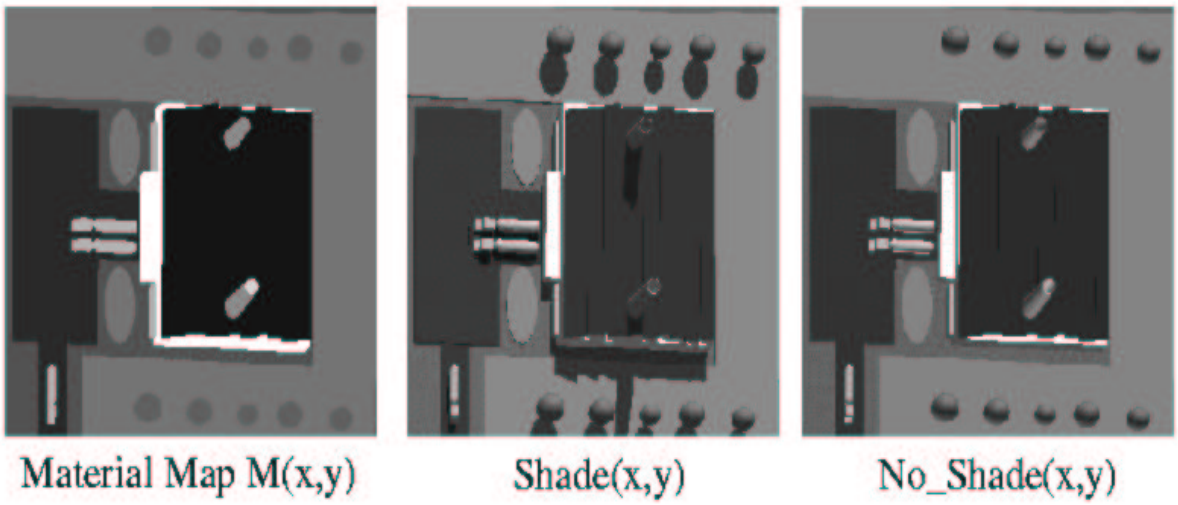

Figure 2. Three raytracing runs with POV generate a materials, illumination and shadow maps

2. Raytracing the scene with all diffuse surface reflectances set to unity and no ambient reflectances resulted in a 8-bit shading image $Shade(x, y)$. The cosine of the angle between the surface normal and the solar vector is then given by: $\cos(\vec{N} \cdot \vec{S}) = Shade(x, y)/255$. A sample is shown in Fig. 2 center, with additional gray levels to show the background surface types.

3. For thermal modeling it is necessary to know if a surface is in shadow or not. Therefore we generate a POV image $NoShade(x, y)$ with the option of turning off mutual shadowing between objects. The shadow map (0 for shadow and 1 for illumination) is given by the difference $Shadow(x, y) = (NoShade(x, y) - Shade(x, y)) > 0$. A sample of $NoShade(x, y)$ with gray levels indicating different materials is shown in Fig. 2 on the right.

2.2. Spectral Model for Surfaces

A number of spectra for natural and man made surfaces were extracted from the Non-conventional Exploitation Factors Data Systems (NEFDS) * and from Salisbury and d'Aria, 1992 †. The emissivity spectra were interpolated to the wavelengths posted by the MODTRAN 3 ‡ run. In order to avoid storing an emissivity data cube we used

*The NEFDS home page is *ciks.cbt.nist.gov/nef/nefhome.html*

†Contact salisburys@worldnet.att.net

‡MODTRAN is available at: www.plh.af.mil/VSBM/gpoc/modtran.html

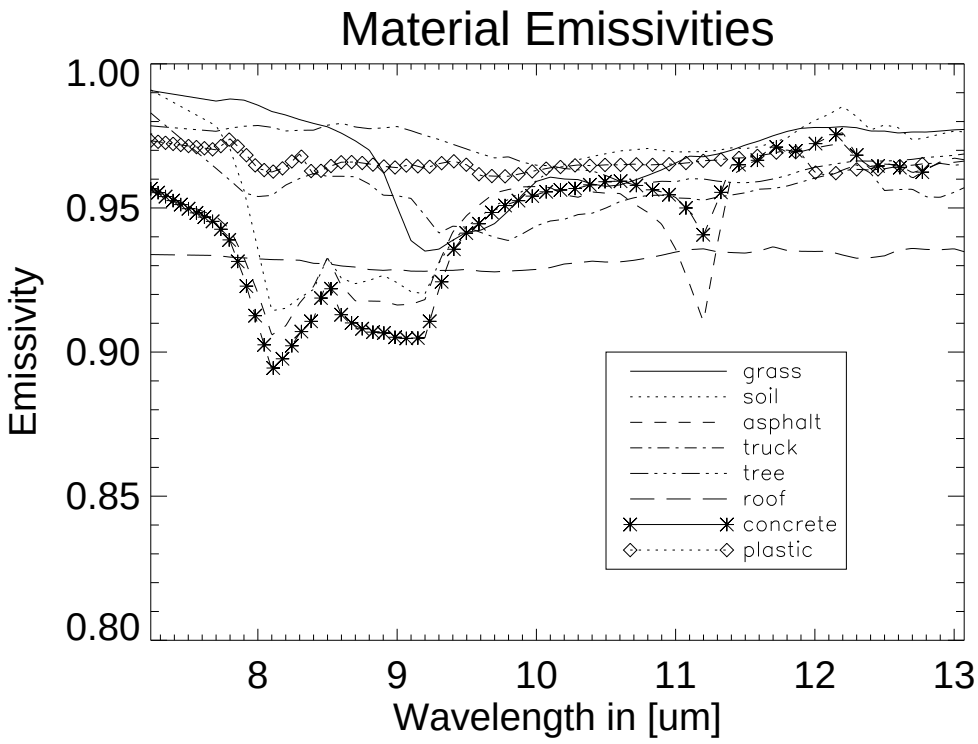

Figure 3. Material emissivities used in simulation of a thermal hypercube.

a materials map $Mat(x, y)$ with values of $n = 1, ..., N_{mat}$. Each number is then used as an index to a library of emissivity spectra $\varepsilon_\lambda(n)$. Fig. 3 shows the emissivities used.

2.3. Thermal Model

Two thermal scene generation programs were developed to compute the surface temperature for the computer-generated scene. The first one is based on statistics derived from field experiments and yields temperature distributions which may not be accurate in detail but is easy to implement.

2.3.1. Statistical Model

Jacobs, 1995, published statistical results on mean day and night-time brightness temperatures for 3 seasons (winter, spring and summer) in the Netherlands. We used these to normalize computed diurnal cycles of solar irradiance on the facility. If a material could not be found in the statistical data it was assumed to have a temperature between two reasonably selected materials. Guesses were made on how much heat is retained by the different surface types between time steps in sunlit and shaded regions. The result of this step is a ground temperature image $T_{ground}(x, y)$.

The algorithm contains the following steps:

1. Compute normalized diurnal cycles of solar irradiance $I_{sun}(n)$. * Cloudiness is modeled as a binary random process multiplied with the fraction of total irradiance as a function of indirect ($f_{indirect}$) and direct ($1 - f_{indirect}$):

$$f_{total}(n) = f_{indirect} + (1 - f_{indirect})R(n),$$

where $R(n) = (r(n) > C_{fraction}(n))$ with $r(n)$ is a random variable with uniform distribution between 0 and 1 and $C_{fraction}(n)$ is the cloud fraction threshold which can be varied as a function of time to simulate the increase of afternoon cloudiness.

*We used the IDL routine called zensun.pro from package esrgidl3.4 by Dr. Paul Ricchiazzi, Earth Space Research Group, UCSB (*skua.crseo.ucsb.edu/esrg.html*).

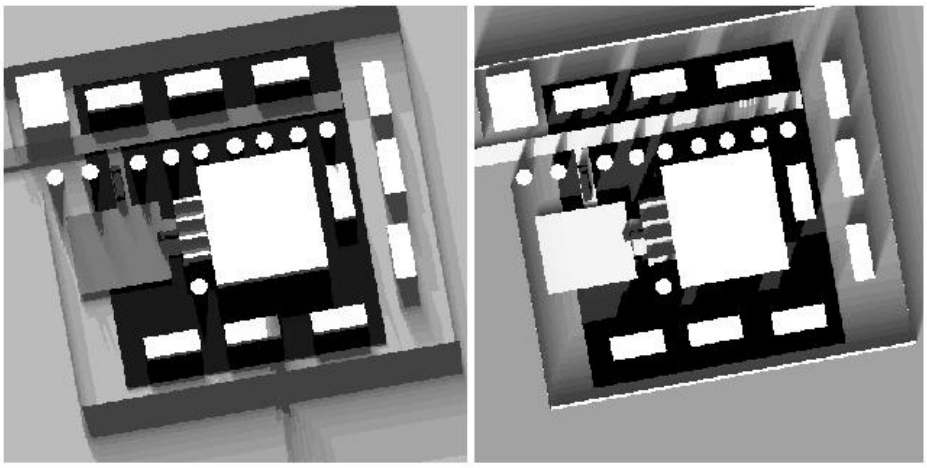

Figure 4. Simulated temperatures for the morning (left) and evening (right).

2. Assign each surface a time constant to simulate heat storage.

3. Compute the ground temperature $T_{ground}(x, y, n)$ at image coordinates x and y at time n as:

$$T_{ground}(x, y, n) = A + B \sum_{i=0}^{n} f_{total}(n) I_{sun}(n) Shade(x, y, n-i) \exp[-\alpha(M(x, y))(n-i)], \quad (1)$$

where $A = T_{night}$ and $B = (s(x, y, n) - min(s(x, y, n)))/(max(s(x, y, n)) - min(s(x, y, n)))$ with $s(x, y, n)$ as the sum in eq. (1). A and B are constants determined to so that the temperature history agrees with the statistically measured temperature range and $\alpha(M(x, y))$ is the material property dependent time constants.

In Fig. 4 we show two examples which have been enhanced using histogram equalization to show small temperature differences. The morning temperature image (on the left) shows sharp shadows because the ground is just starting to warm up after the night. The evening temperature image - just at sunset - (on the right) shows washed-out shadows indicating the effect of gradual cooling in the shadowed regions as the Sun changes its position. While this method generates realistic temperature ranges for various surface types and includes shading histories, the simple model does not include the effect of heat storage below the surface which is important for several hours after sunset occurs.

2.3.2. Finite-Difference Model

To model the solar irradiance, heat conduction and turbulent heat exchange with the atmosphere a finite difference model was developed based on equations found in heat transfer textbooks, e.g. Incropera and De Witt, 1985. The energy balance equations states that:

$$q_{rad} + q_{wind} + q_{cond} = E_{storage},$$

where for the n-th time-step:

$$q_{rad} = \varepsilon_T \sigma (T_{sky}^4 - T_{0,n}^4) + \varepsilon_V Q_{sol},$$

$$q_{wind} = h_{wind} v_{wspeed}^{0.8} (T_{wind} - T_{0,n}),$$

$$q_{cond} = k \frac{T_{0,n} - T_{1,n}}{\Delta z}$$

and

$$E_{storage} = \rho C_p \frac{\Delta z}{2} \frac{T_{0,n+1} - T_{0,n}}{\Delta t}.$$

Material	Asphalt	Soil	Wood	Concrete	Brick	Copper	Plastic	Steel
ρ	2115.	1300.	530.	2300.	2000.	8400.	1800.	7920.
k	0.062	0.8368	0.129	1.046	1.0042	230.12	0.2092	13.8
C_p	920.	1046.	2301.	656.9	753.	376.6	1255.	460.2
ε_V	0.83	0.68	0.84	0.60	0.63	0.82	0.52	0.80
ε_T	0.85	0.93	0.94	0.90	0.95	0.33	0.85	0.27

Table 1. Thermal properties of eight materials (adapted from Incropera and De Witt, 1985).

The temperature of the layer below the surface is given by:

$$T_{1,n} = T_{1,n-1} + \frac{2q_0}{k}\sqrt{\frac{\alpha\Delta t}{\pi}}\exp[-\frac{\Delta z^2}{4\alpha\Delta t}] - \frac{q_0\Delta z}{k}erfc(\frac{\Delta z}{\sqrt{4\alpha\Delta t}}),$$

where $q_0 = q_{rad} + q_{wind}$, $\sigma = 5.67x10^{-6}W/m^2K^4$ is the Stefan-Boltzman constant, h_{wind} is 4.786 for a turbulent flow, v_{wspeed} is the windspeed in m/s, $\alpha = k/(\rho C_p)$ is the thermal diffusivity in m^2/s, $erfc(x) = 1 - erf(x)$ is the complimentary error function, k is the thermal conductivity in $W/(mK)$, ρ is the density in kg/m^3, C_p is the specific heat in $J/(kgK)$, ε_x is the emissivity in the visible ($x = V$) and thermal ($x = T$), Δt is the time step in s and Δz is the layer thickness in m.

In Tab. 1 we list typical thermal and optical properties for several materials of interest:

Combining the above equations and solving for the temperature of the surface layer 0 we obtain:

$$T_{0,n+1} = T_{0,n} + \frac{2\Delta t}{\rho C_p \Delta z}[q_{rad} + q_{wind} + q_{cond}]. \quad (2)$$

In Fig. 5 we show the output of the model for the materials listed in Tab. 1. The wind speed was 4 m/s, the air temperature 5 deg C, the irradiance was set for Julian day 88 and a latitude of 34 deg.

2.4. Gas Plume Simulation

We are interested in simulating thin gas plumes in our hyperspectral scenes to investigate various methods of finding the plume against a complex background, identifying the gas or gases and quantifying the amount of gas.

There are some codes available to realistically simulate the transport of particles under specific atmospheric conditions, e.g. the Monte Carlo model of Blackadar, 1997. Unfortunately it is necessary to track many particles to produce useful gas concentration maps.

For the purpose of plume simulation in scene generation we need a model which produces three dimensional gas concentration distributions. The Gaussian plume model, e.g. Beychok, 1994, can be used for scenes where the spatial resolution of the simulated sensor is poor and/or many measurements are averaged.

For higher spatial resolutions and to simulate scenes for sensors which acquire a spectrum over some time (e.g. FTIR) it is necessary to model turbulent features in the plume. We developed a simple method to generate realistic plumes using time-dependent multi-fractals. The basic principle is that a randomly generated two-dimensional, time-dependent "brush" is scaled, rotated and translated along multiple fractal generated paths in 3-D space. We needed to generate a time-dependent plume to eventually study the effect of time-variable plume transmission on a staring imaging FTIR. The "brush" (2-D) and paths(1-D) are animated by changing the phase of the spectral filter which is used to generate the fractal. An example of a 1-D time-dependent fractal is shown in Fig. 6.

In Fig. 7 we display an enhanced sequence of images to show the shifting, scaling, rotating, and altering of the fractal dimension as a function of plume height. The image sequence starts in the upper left (UL) corner and progresses in raster fashion from left to right to the lower right (LR) corner. In the plume model we conserve the total amount of gas on each horizontal slice. When we render the plume we must also consider the 3-D plume temperature distribution for which we have found no models in the literature. Due to the lack of models and to keep

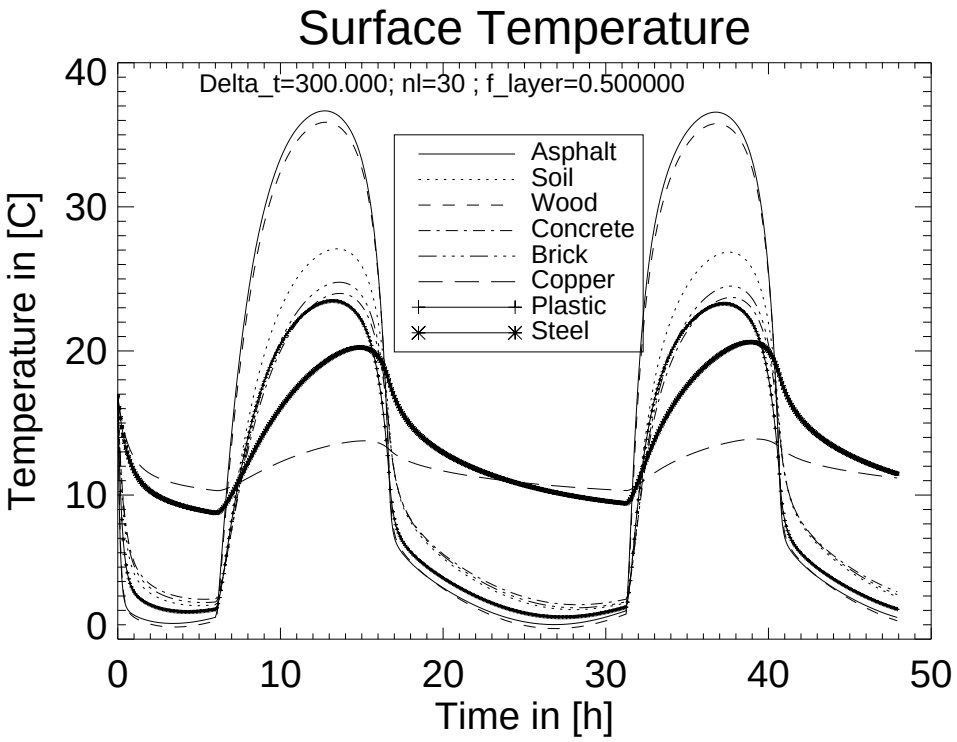

Figure 5. Surface temperatures for two diurnal cycles for eight materials.

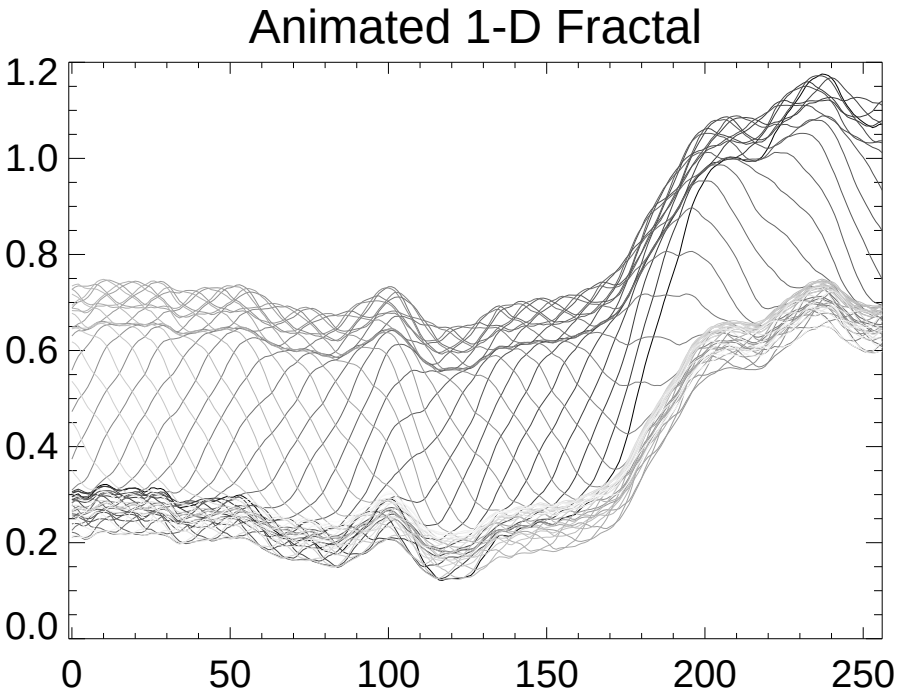

Figure 6. Example of a 1-D time-dependent fractal - the curves are shaded according to the time variable

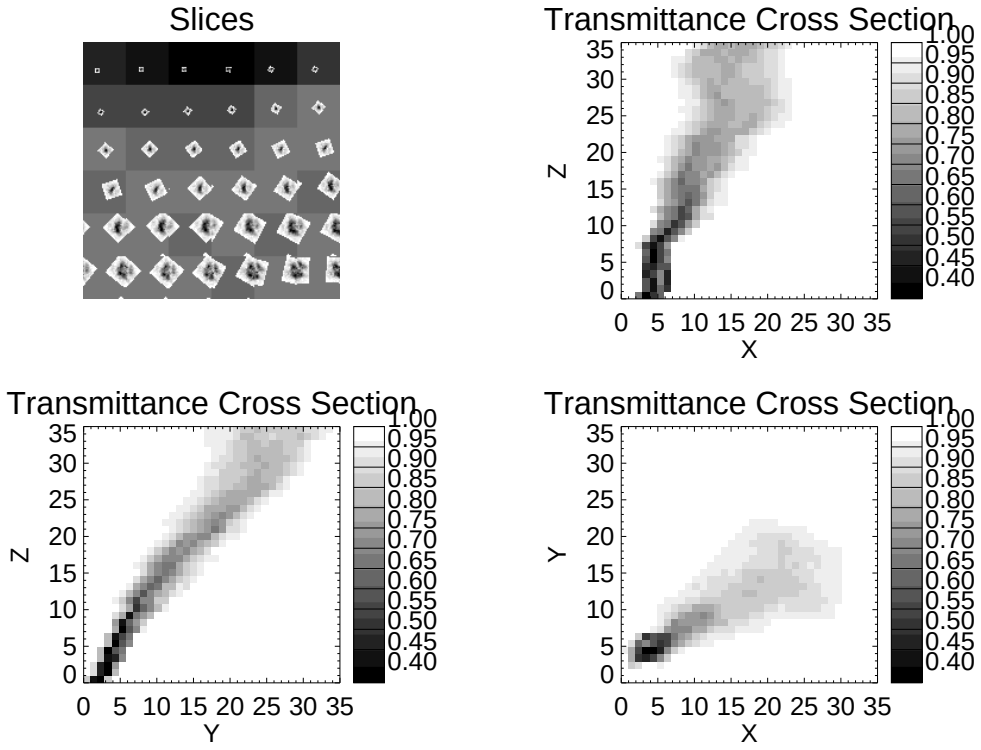

Figure 7. Time sequence of horizontal plume slices demonstrating wind effects, diffusion (scale), plume rotation and changing fractal dimensions (UL). projected plume transmission in x/z (UR), y/z (LL) and x/y (LR) planes.

the calculation simple we assume the effective radiative plume temperature difference at a point x,y is proportional to one minus the transmission which is a function of the gas density:

$$T_{gas}(x,y) = T_{air} + \Delta T \frac{1 - \tau_{gas}(\lambda_{absorb})}{max(1 - \tau_{gas}(\lambda_{absorb}))},$$

where ΔT is a scaling factor, T_{air} is the air temperature and $\tau_{gas}(\lambda_{absorb})$ is the plume transmission at the maximum absorption at wavelength λ_{absorb} of the gas in the spectral range of interest. To perform the calculation exactly, line integrals (e.g. Borel and Gerstl, 1991) would have to be computed through each point of the plume as a function of wavelength.

2.5. Atmospheric Model

We used MODTRAN 3 with an IDL graphical user interface developed by D. Schlaepfer *, 1995 to generate input files (tape5), run MODTRAN and reduce the output files (tape6 and tape7) to up-/down-welling path radiances and total transmission.

2.6. Data-cube Generation

The temperature image $T_{ground}(x,y)$ was then used with emissivity map $\varepsilon(x,y,\lambda) = \varepsilon_\lambda(Mat(x,y))$ for each surface type. The atmospheric transmission $\tau_{atmo}(\lambda)$ and up-/down-welling path radiance ($L_{path\uparrow}(\lambda)$, $L_{path\downarrow}(\lambda)$) data was computed by MODTRAN 3 to generate a radiance image cube:

$$L_{total}(x,y,\lambda) = L_{ground}(x,y,\lambda) + L_{gas}(x,y,\lambda) + L_{path\uparrow}(\lambda) + L_{reflected}(x,y,\lambda), \quad (3)$$

*MODO is available as shareware through FTP from *ftp.geo.unizh.ch/pub/dschlapf/Mod_Util/* or by request from the author dschlapf@geo.unizh.ch

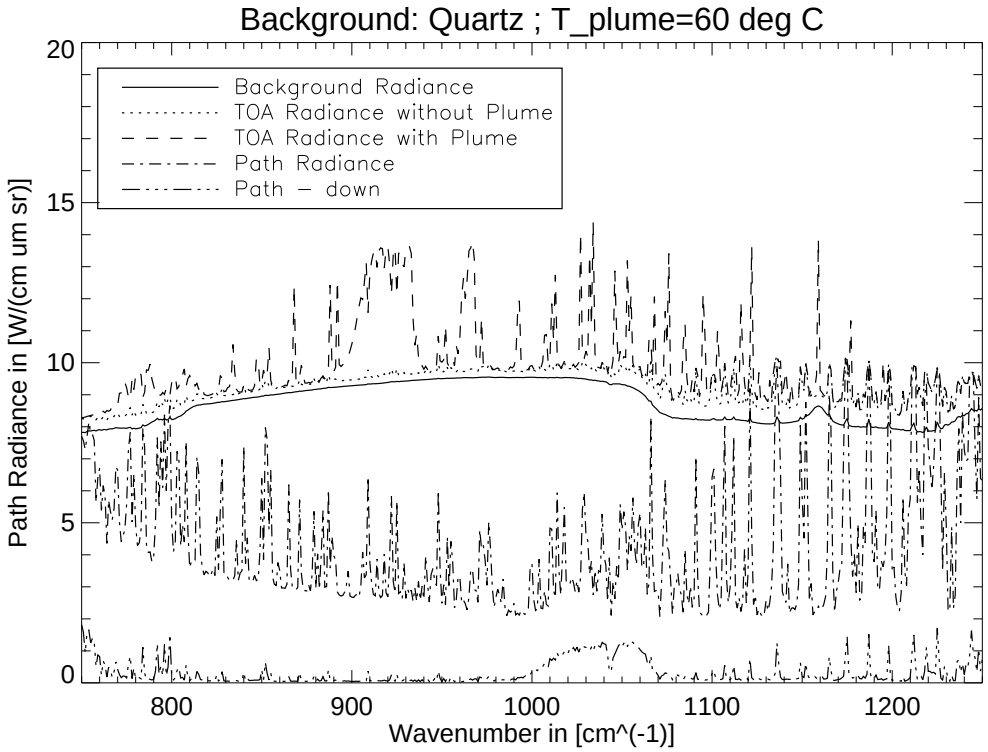

Figure 8. Radiances computed over quartz.

where

$$L_{ground}(x,y,\lambda) = \varepsilon(x,y,\lambda)B(\lambda,T_{ground}(x,y))\tau_{gas}(x,y,\lambda)\tau_{atmo}(\lambda),$$
$$L_{gas}(x,y,\lambda) = [1-\tau_{gas}(x,y,\lambda)]B(\lambda,T_{gas})\tau_{atmo}(\lambda),$$
$$L_{reflected}(x,y,\lambda) = L_{path\downarrow}(\lambda)[1-\varepsilon(x,y,\lambda)]\tau_{gas}(x,y,\lambda)\tau_{atmo}(\lambda),$$

and $B(\lambda,T)$ is the Planck function describing the spectral radiance in $[W/(cm^2\ ster\ \mu m)]$. For stacks over highly reflective surfaces it might be necessary to include a term for the down-welling radiance from the gas plume. In Fig. 8 we show how a 60 deg C gas plume with 500 ppm-m ammonia and 450 ppm-m tetrachloroethylene (TCE) appears over quartz background at 28 deg C. [†]. The effect of resolution on the ability to detect a gas plume is shown in Fig. 9. The plume temperature was set at 45 deg C and the background was quartz. The relative radiances with the plume are drawn with solid lines for 2 cm^{-1} resolution at the bottom and 42 cm^{-1} at the top in steps of 4 cm^{-1}. Notice that the narrow absorption features of ammonia disappear quickly and only the TCE absorption near 920 cm^{-1} is visible until the resolution is greater than 30 cm^{-1}. Using eq. (3) a spectral cube was generated and visualized using 3-D image cube visualization ENVI in Fig. 10.

The generation takes about 30 sec for a $N_x \times N_y \times N_{chan} = 128 \times 128 \times 128$ (5 cm^{-1} sampling) cube and 8 min for a $320 \times 320 \times 751$ (1 cm^{-1} sampling) cube on a SGI Indigo2 with a R8000 64-bit processor running at 75 MHz.

3. TEMPERATURE-EMISSIVITY SEPARATION (TES) ALGORITHM

In order to analyze hyperspectral thermal imagery it is necessary to separate the surface temperature and emissivity. Several methods exist to perform TES for multi-spectral sensors, e.g. Hook et all, 1992, Matsunaga, 1993 and Gillespie et al., 1996 [*]. Looking at spectral data bases (e.g. Salisbury and D'Aria, 1992) we find that typical surface emissivity spectra are rather smooth compared to spectral features introduced by the atmosphere. This fact is used in the TES method we have developed (see Borel, 1998) and can be summarized by the following steps:

[†]High resolution gas spectra can obtained from EPA's Emission Measurement Center (*134.67.104.12/html/emticwww/ftir.htm*) and Galactic Industries Web site (*www.galactic.com*).

[*]The document is available on the WEB at *http://asterweb.jpl.nasa.gov/asterhome/docs.htm.*

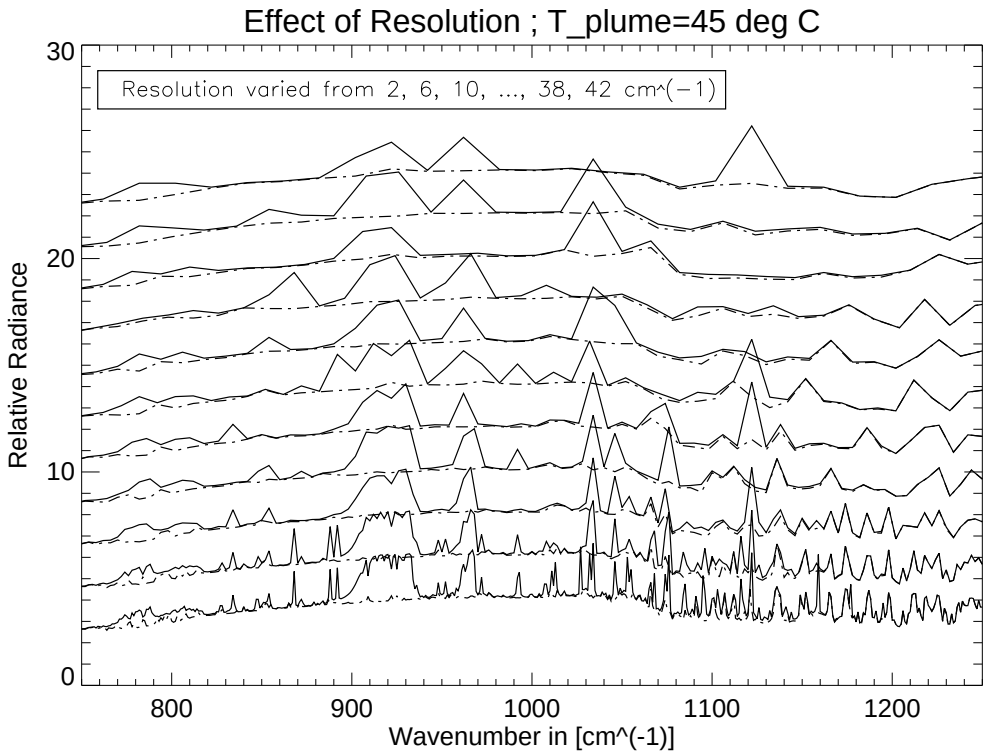

Figure 9. The effect of spectral resolution on the observed on-plume (solid line) and off-plume (dashed line) radiance.

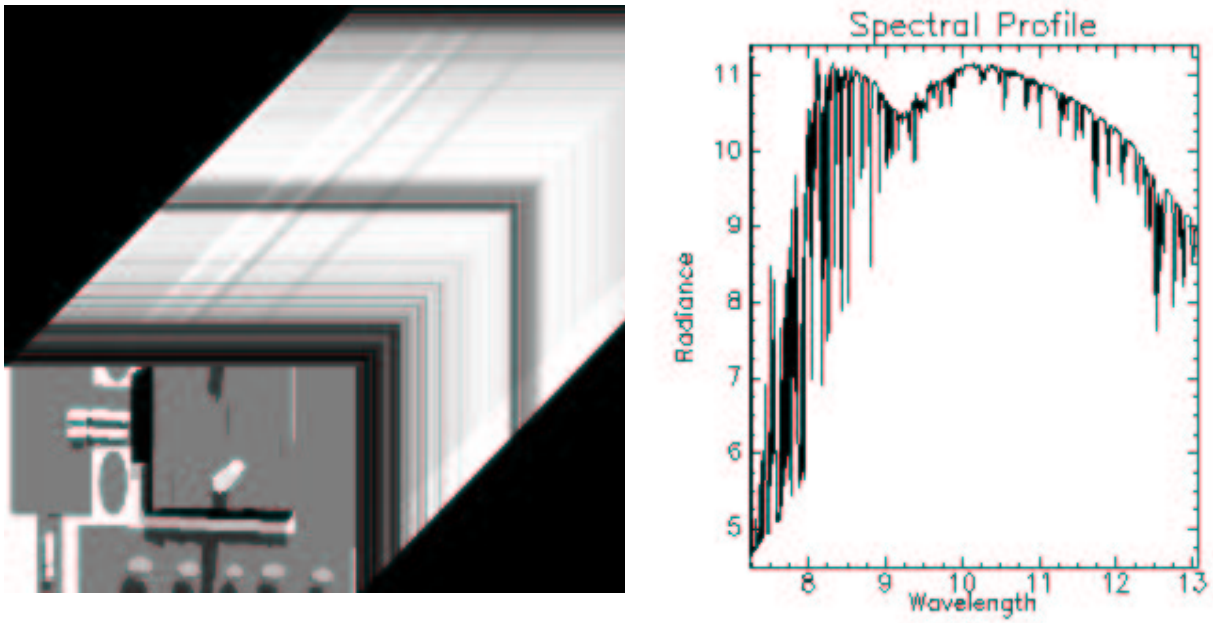

Figure 10. Simulated thermal hypercube.

1. Compute the initial ($n = 0$) blackbody temperature $T_{bb,n}$ in an atmospheric window from an atmospherically corrected radiance $L_{cor,0}$:

$$T_{bb,n} = B^{-1}\left(\lambda_{window}, L_{cor,n}\right)$$

with

$$L_{cor,n} = \frac{L_{total} - L_{path\uparrow}(CW, T_{atmo}) - L_{path\downarrow}\varepsilon(n)}{\varepsilon(n)\tau_{atmo}(CW)},$$

where CW stands for column water, T_{atmo} is the effective atmospheric temperature and we begin the iterations assuming an average emissivity $\varepsilon(0) = 0.95$ which is a good approximation for wavelengths from 11.5 to 12 μm.

2. Compute spectral emissivity: $\varepsilon(n) = L_{cor,n}/B(\lambda, T_{bb,n}), n = 1, 2, ..$

3. Vary the surface temperatures $T_{bb,n} = T_{bb,0} + i\Delta T,\ i = 1, 2, ...,$ change the columnar water amounts and the effective atmospheric temperatures and recompute $\varepsilon(n)$ iteratively using steps 1-3.

4. Stop iteration when emissivity is smoothest, i.e. when

$$\sigma(\varepsilon(n)) = STDEV\left[\varepsilon_i(n) - \frac{1}{K}\sum_{j=i-K/2}^{i+K/2-1}\varepsilon_j(n)\right]_{i=K/2+1,\ldots,M-K/2} = Min,$$

where the spectrum consists of M channels.

4. CONCLUSIONS

We have presented a methodology to generate hyperspectral cubes which include realistic temperature distributions, material dependent emissivities, complex gas plumes and atmospheric absorption and emission. Many improvements are possible, e.g. consider radiative heat exchange between surfaces, calculation of local wind speeds, physically based 3-D plume dispersion models, include other thermal models to calculate surface temperatures of thin layers, mixture models for emissivities.

References

Beychok, M. R., *Fundamentals of stack gas dispersion*, Irvine, 223 Martin St., Unit 205, CA 92612, 193 p., 1994.

Borel C. C. and S. A. W. Gerstl, "Simulation of partially obscured scenes using the radiosity method", SPIE Technical Symp. on Opt. Eng. and Phot. in Aerosp. Sens., Orlando, FL, Vol. 1486, pp. 271-277, 1991.

Borel, C. C., "Surface emissivity and temperature retrieval for a hyperspectral sensor", IGARSS'98, Seattle, WA, July 6-10, 1998.

Blackadar, A. K., *Turbulence and diffusion in the atmosphere*, Springer, 185 p., 1997.

Gillespie, A. R., S. Rokugawa, S. J. Hook, T. Matsunaga, and A. B. Kahle, "Algorithm theoretical basis document for temperature/emissivity separation", Version 2.3, 16 August 1996.

Hook S. J., A. R. Gabell, A. A. Green and P. S. Kealy, "A comparison of techniques for extracting emissivity information from thermal infrared data for geologic studies," *Remote Sens. Environ., 42*, pp. 123-135, 1992.

Incropera, F. P. and D. P. De Witt, *Fundamentals of heat and mass transfer*, John Wiley & Sons, 802 p., 1985.

Jacobs, P., "Thermal infrared characterization of ground targets and backgrounds," pp. 199-200, SPIE Tutorial Texts in Optical Engineering, 1996.

Matsunaga, T., "An emissivity-temperature separation technique based on an empirical relationship between mean and range of spectral emissivity," *Proc. 14th Japanese Conf. of Remote Sensing*, pp. 47-48, 1993.

Salisbury, J. W. and D. M. D'Aria, "Emissivity of terrestrial materials in the 8-14 μm atmospheric window," *Remote Sens. Environ., 42*, pp. 83-106, 1992.